

**Information Systems Division**  
**Measurement Program**  
**Implementation Plan**

**Version 1.0**

**August 19, 2005**

# TABLE OF CONTENTS

<b>1</b>	<b><u>INTRODUCTION</u></b>	<b>3</b>
1.1	<u>Purpose</u>	3
1.2	<u>Goals</u>	3
1.3	<u>Governing Policy and Scope</u>	3
<b>2</b>	<b><u>IMPLEMENTATION APPROACH</u></b>	<b>4</b>
2.1	<u>Background</u>	4
2.1.1	<u>Leverage Ongoing Measurement Efforts</u>	4
2.1.2	<u>Implement Measurement at Two Levels</u>	4
2.2	<u>Overall Implementation Strategies</u>	5
2.3	<u>Phasing Strategies</u>	5
<b>3</b>	<b><u>IMPLEMENTATION ELEMENTS</u></b>	<b>5</b>
3.1	<u>ISD Product and Process Measurement Definition Work Element</u>	6
3.2	<u>ISD Measurement Collection Procedures Work Element</u>	6
3.2.1	<u>Data Gathering for Models and Trends</u>	7
3.2.2	<u>Data Gathering for Assessing Compliance with the ISD Process Baseline</u>	8
3.3	<u>ISD Measurement Analysis Procedures Work Element</u>	8
3.3.1	<u>Analyzing Models and Trends</u>	9
3.3.2	<u>Analyzing Process Compliance</u>	1
3.4	<u>ISD Measurement Program Deployment Element</u>	2
<b>4</b>	<b><u>ISD MEASUREMENT PROGRAM INTERFACES AND ORGANIZATIONAL STRUCTURE</u></b>	<b>3</b>
<b>5</b>	<b><u>REVIEW, REPORTING, AND METRICS</u></b>	<b>5</b>
<b>6</b>	<b><u>PROJECT SCHEDULE</u></b>	<b>5</b>
<b>7</b>	<b><u>BUDGET</u></b>	<b>5</b>
<b>8</b>	<b><u>USE OF EXTERNAL ORGANIZATIONS</u></b>	<b>5</b>
<b>9</b>	<b><u>RISK MANAGEMENT</u></b>	<b>5</b>
<b>APPENDIX A:</b>	<b><u>Gathering Organizational Measures</u></b>	<b>7</b>
<b>APPENDIX B:</b>	<b><u>Gathering Compliance Measures</u></b>	<b>13</b>
<b>APPENDIX C:</b>	<b><u>Analyzing Models and Trends</u></b>	<b>14</b>
<b>APPENDIX D:</b>	<b><u>Analyzing Compliance Measures</u></b>	<b>23</b>
<b>APPENDIX E:</b>	<b><u>ACRONYMS AND ABBREVIATIONS</u></b>	<b>24</b>

## **1 Introduction**

The success of Goddard Space Flight Center (GSFC) projects has become increasingly dependent on the quality and reliability of software. Budget caps, shorter development times, and increased demands on software capabilities have raised concerns of mission software quality, safety and developmental predictability. GSFC is committed to measurably improving software development practices to improve schedule and budget predictability and minimize risks attributed to software defects.

GSFC has expanded its ongoing software process improvement efforts by forming a Software Process Improvement (SPI) Project under the Information Systems Division (ISD). The project's goal is to establish and continuously improve system and software products and processes by providing the necessary supporting infrastructure including metrics collection and analysis. Several teams have been formed to address key SPI programmatic areas, including the Engineering Process Group (EPG), ISD Process Team, and ISD Measurement Team.

### **1.1 Purpose**

This document describes the plan for one program area under the SPI Project, the ISD Measurement Program. The purpose of this plan is to describe the ISD Measurement Program's goals and implementation approach. The plan addresses the scope, implementation approach, operations concept, organization structure, schedule, and budget. The plan will be revised, as necessary, in response to evolving implementation needs and constraints.

### **1.2 Goals**

The goals of the ISD Measurement Program are to:

- Build software models for use by future projects (e.g., estimation models for effort, schedule, defects)
- Track performance trends (e.g., requirements volatility, development predictability, productivity)
- Assess the impact of SPI Project efforts on ISD project performance
- Provide measurement support to projects as requested
- Provide support to the ISD and its projects in meeting NASA measurement requirements (e.g., NPR 7150.2)

### **1.3 Governing Policy and Scope**

The Chief of the ISD has established a policy (ISD Software Policies, Version 1.094, April 1, 2005) that ISD software development projects are required to provide measurement data to the ISD Measurement Team.

The scope of this policy includes all mission software projects (Class A, B or C as defined in NPR 7150.2) in the ISD, as well as other ISD software projects that are estimated at more than 5 staff-years effort at their inception. Software project managers may request a waiver to this policy. The ISD Division Chief or a designee has authority to approve waivers.

## **2 Implementation Approach**

This section describes the overall implementation strategies and work implementation elements. Background information on how the GSFC SPI Project applies past experience to guide the deployment of the ISD Measurement Program is also presented.

### **2.1 Background**

The ISD Measurement Team of the GSFC SPI Project investigated the current activities of the JPL Software Quality Initiative (SQI) Project and the past work of the GSFC Software Engineering Laboratory (SEL) to gather lessons learned. This information was used to help the ISD Measurement Team develop strategies, while avoiding past problems.

#### **2.1.1. Leverage Ongoing Measurement Efforts**

Currently, there are several measurement activities within ISD to leverage:

- Tracking and analyzing defects, test performance and other parameters
- Using progress points to track and analyze progress

The SPI Project plans on enhancing measurement activities within the ISD to make measurement into a valuable, compliant asset for ISD projects and management.

#### **2.1.2. Implement Measurement at Two Levels**

Based on lessons learned, most recently at JPL, the SPI Project is implementing two thrusts in software measurement for ISD:

- First: at the ISD level for organization-wide perspective
- Second: at the 'Project' level to address needs of specific projects or branches

Each thrust is supported by activities within the SPI Project Teams:

- At the ISD-level, for future project support
  - Develop experienced-based aids to support all ISD projects (e.g., cost and defect models)
  - Determine if ISD software development processes are applied and are useful

- Identify endemic software-related issues in the organization (e.g., types of defects, requirements growth)
- At the project-level, for project management
  - Provide insight into project status and performance (e.g., Will I meet commitments? Is there some reason we are falling behind?)
  - Provide guidance in selecting technologies (e.g., Should I be applying inspections? Structural testing? Use Matlab for code generation?)
  - Satisfy NASA software engineering requirements

## **2.2 Overall Implementation Strategies**

In addition to the principles described in the previous section, the ISD Measurement Program will use the following key implementation strategies:

- Work with ISD management to provide product and process analyses to support their strategic decision-making.
- Work with the software leads and SPI Project to provide guidance on efficient and effective project-level measurement.
- Actively engage key partner project personnel. Specifically, Phase I of the ISD Measurement Program will engage flight projects and mission-critical software applications in the Flight Software Branch (FSB).
- Operate as a program under the SPI Project. The ISD Measurement Program will have an integrated schedule, receivables/deliverables, performance metrics, and monthly and annual reviews.

## **2.3 Phasing Strategies**

The ISD Measurement Program will employ a phasing strategy to minimize risks associated with broad-based development and deployment.

- Phase I focuses on developing and deploying software management measurement models and performance trends for the FSB and/or its projects in the ISD.
- Phase II builds on experiences gained from the initial deployment in the FSB. Also, it focuses on refining and extending the set of software engineering measurement models to include other ISD Branches and projects.
- Phase III adds process compliance measures and trends. Proposed elements of this phase are included in the plan as placeholder information and annotated as such.

## **3 Implementation Elements**

The activities and products of the GSFC SPI Project and ISD Measurement Program will contribute to satisfying the SPI goals of each ISD Branch as well as NASA measurement goals. The ISD essential measurement and associated improvement goals are driven by delivery of quality products on cost and schedule.

The ISD Measurement Program is organized around four work elements that integrate to achieve the overall program goals. These elements are:

- ISD Product and Process Measurement Definition
- ISD Measurement Collection Procedures
- ISD Measurement Analysis Procedures
- ISD Measurement Program Deployment

### **3.1 ISD Product and Process Measurement Definition Work Element**

The main function of this work element is to document measures to satisfy essential ISD measurement goals.

The principal approaches of this work element include:

- Working with Project teams to integrate NASA measurement needs into project-level processes
- Defining interfaces with related ISD implementation processes (e.g., Bimonthly Status Reviews, quality assurance organizations)
- Defining a set of measures and related analysis and assessment techniques for ISD-level modeling and trending
- Using the Software Engineering NPR 7150.2 and the NASA measurement and analysis requirements (e.g., CMMI) as a framework for describing project measures

The primary products from this work element are preliminary measurement data definitions, model and trend examples, and assessment examples.

The products generated in this work element will be updated, based on feedback from project users and the Deployment work element. Additional products may be generated, as necessary.

### **3.2 ISD Measurement Collection Procedures Work Element**

The main functions of this work element are to provide a description of the operational scenarios, collection frequency and supporting infrastructure for collecting ISD measures. These measures will be used to develop models and trends as well as assess compliance against the ISD Process Baseline.

The principal approaches of this work element include:

- Establishing a procedure to define the information to be collected and the frequency of collection and an information repository
- Providing tools and methods for data collection and storage

### 3.2.1. Data Gathering for Models and Trends

Measures for models and trends are provided to the measurement team by the product development leads (PDLs) from measurement data reports as well as project start and end interviews. The PDL and a measurement team member will meet at project start to establish the measures to be provided by the PDL and the schedule and method(s) to be used. During development, the PDL provides data on an agreed-to schedule via tools supplied by the measurement team; additional meetings may be requested by the PDL. At closeout, there is a wrap-up meeting with the PDL to collect final data. Measurement reports can be provided periodically or at key review points, but need to include data from, at least, major milestones. Because the elapsed time between reviews can be several months, bimonthly status materials may also be gathered as needed. The types of data and the times they are collected are:

1. *Project characteristics data* are collected at the start of a project and include basic information such as what type, subtype of software (e.g., flight software, ACS) is being developed, what languages are being used, and point of contact information.
2. *Project planning data* are collected at the start of a project, at major milestones, and in response to major project re-planning. The minimal set of milestones at which the data are collected includes project start, PDR, CDR, and completion of acceptance test. These data include planned effort, progress points, and milestones dates. The basis of estimate will also be collected initially and at project re-plans.
3. *Periodic status (“actuals”) data* are collected throughout the project as agreed to at project start. These data include effort spent and progress made at milestones. Project dynamics data relating to defects and requirements volatility are also collected.
4. *Development completion statistics* are collected when software enters the maintenance phase. This usually but not always entails a turnover from the development team to a maintenance team. These data include the final size by subsystem/CSCI, milestone dates, and effort data. The effort data include total effort and a breakdown by phases.

The primary products from this work element are procedures and operational scenarios for data gathering and storage.

The procedure in Appendix A, “Gathering Organizational Measures,” describes the details of what data are needed and how the ISD Measurement Team acquires this software engineering information from projects.

### 3.2.2. Data Gathering for Assessing Compliance with the ISD Process Baseline <sup>1</sup>

In Phase III, measures for the process compliance will be based on adherence to the defined ISD process baseline as tailored for the local project and organization. These measures will be gathered by way of ‘Quick Look Assessments (QLA)’ which will be executed by the Quality Assurance organization. All projects in ISD will participate in these assessments.

The characteristics of these QLAs are that they are based on sampling a subset of target practices as opposed to all required practices. They are designed to require a minimum of overhead to each project in that each audit will typically take only two meetings and will typically require less than three hours of time from each participating team member from the project. Additionally, the audit inquiries are kept at a very high level as opposed to the detail typically used by many benchmark audits. For example, the audit team would determine if inspections are being done; not whether detailed records of the inspections are archived.

The QLAs are carried out by the Quality Assurance organization and they are supported by staff from the SPI Project as needed. There are several instruments used to support the process measures and the QLAs. These artifacts include:

- Defined QLA process
- Process compliance scoring spreadsheet
- QLA report form

The process measures are derived directly from the QLA scoring spreadsheet.

The procedure in Appendix B, “Gathering Compliance Measures,” describes the details of how this software process assessment information is collected from projects.

The primary products from this work element are procedures and operational scenarios for data gathering and storage.

The products generated in this work element will be updated, based on feedback from project users and the Deployment work element. Additional products may be generated, as necessary.

### **3.3 ISD Measurement Analysis Procedures Work Element**

The main functions of this work element are to provide a description of the usage and analysis of the collected ISD measures and the development of models and trends as well as compliance assessments against the ISD Process Baseline. Current basis of estimate (BOE) parameters and historical data may be used on a limited basis to provide initial values for the models. Sources for these data include past project reports and/or historical data (e.g., SEL database).

---

<sup>1</sup> Note: this section describes a proposed approach for Phase III process compliance measurement and needs further development.



The principal approaches of this work element include:

- Analyzing past project reports and historical data sources for initial model parameters
- Producing and using models and trend reports from the ISD measures
- Analyzing process assessments to produce compliance reports

The data collection and repository structures in Appendix A provide for user designated Software Type and composite Type model formulations, such as ACS flight software, ground planning and scheduling software.

### 3.3.1. Analyzing Models and Trends

Table 1 shows the initial set of models and trends to be developed. At the ISD level, the near-term focus will be on producing planning, estimation and prediction models for use by software leads. Over time, the cross-project trends can provide guidance in making strategic decisions about ISD institutional needs.

**Table 1. ISD Models and Trends**

<b>Model Type</b>	<b>Model Names</b>	<b>ISD Measures Needed to Build Model</b>	<b>Output</b>
Planning: Initial Effort and Schedule Estimation	1. Total effort estimate	<ul style="list-style-type: none"> <li>• Effort (actual at end of development)</li> <li>• Earned value points (actual at end of development)</li> <li>• Size (actual at end)</li> <li>• Software type</li> </ul>	<ul style="list-style-type: none"> <li>• Estimate of total effort and expected range</li> <li>• Estimate of total earned value points</li> </ul>
	2. Effort estimate by phase	<ul style="list-style-type: none"> <li>• Milestone dates (actuals)</li> <li>• Effort (actuals at phases)</li> <li>• Earned value points (actuals at phases)</li> <li>• Software type</li> </ul>	<ul style="list-style-type: none"> <li>• Estimate of effort (expected range) by milestone phase</li> <li>• Estimate of progress points by milestone phase</li> </ul>
	3. Total schedule estimate	<ul style="list-style-type: none"> <li>• Schedule (actual start/stop dates of development)</li> <li>• Software type</li> </ul>	<ul style="list-style-type: none"> <li>• Estimate of total development duration and expected range</li> </ul>
	4. Schedule estimate by phase (phase duration)	<ul style="list-style-type: none"> <li>• Milestone dates (actuals)</li> <li>• Software type</li> </ul>	<ul style="list-style-type: none"> <li>• Estimate of duration (expected range) of phases</li> </ul>
Planning: Remaining Effort and Schedule Prediction	5. Remaining effort prediction	Uses Models 1 and 2 and current Project data: <ul style="list-style-type: none"> <li>• Effort (actuals at phases)</li> <li>• Earned value points (actuals at phases)</li> </ul>	<ul style="list-style-type: none"> <li>• Predicted effort remaining for current project based on actuals to date</li> <li>• Predicted total earned value points</li> </ul>
	6. Schedule tracking and prediction	Uses Models 3 and 4 and current project data: <ul style="list-style-type: none"> <li>• Milestone dates (actual to-date)</li> </ul>	<ul style="list-style-type: none"> <li>• Predicted schedule and milestone dates remaining for current project based on actuals to-date</li> </ul>

<b>Model Type</b>	<b>Model Names</b>	<b>ISD Measures Needed to Build Model</b>	<b>Output</b>
Planning: Requirements Volatility Profile	7. Requirements volatility by phase	<ul style="list-style-type: none"> <li>Requirement changes (i.e., number added, deleted, modified) by phase</li> <li>Number of TBDs by phase</li> </ul>	<ul style="list-style-type: none"> <li>Expected requirements volatility (range) (% requirements change)</li> <li>Expected number of TBDs (range)</li> </ul>
Planning: Defect Profile	8. Defect profile by phase	<ul style="list-style-type: none"> <li>Number of defects found in configured software library by severity by phase</li> <li>Size at end of development</li> </ul>	<ul style="list-style-type: none"> <li>Expected number of defects by phase</li> <li>Defect density (# defects/size)</li> </ul>
Trending: Cross-project Indicators	9. Productivity trend	<ul style="list-style-type: none"> <li>Same data as Models 1 and 2</li> </ul>	<ul style="list-style-type: none"> <li>Productivity = unit of value (size, points)/ unit of effort</li> </ul>
	10. Management performance trend	Same data as Models 5 and 6 with additional data: <ul style="list-style-type: none"> <li>Estimates of milestones and estimates of effort</li> </ul>	Management performance = <ul style="list-style-type: none"> <li>% deviation of effort estimate from actual effort</li> <li>% deviation of schedule estimate from actual schedule</li> </ul>
	11. Impact of Requirements Volatility	<ul style="list-style-type: none"> <li>Same data as Models 7, 8 and 9</li> </ul>	<ul style="list-style-type: none"> <li>Show impact of requirements volatility on quality and productivity</li> </ul>
Trending: SPI Indicators	12. Productivity by CMMI or internal assessment level	<ul style="list-style-type: none"> <li>Assessment rating</li> <li>Same data as Model 9</li> </ul>	<ul style="list-style-type: none"> <li>ROI by CMMI level</li> </ul>
	13. Defects by CMMI or internal assessment level	<ul style="list-style-type: none"> <li>Assessment rating</li> <li>Same data as Model 8</li> </ul>	<ul style="list-style-type: none"> <li>Reliability by CMMI level</li> </ul>
	14. Effects of technology or process	<ul style="list-style-type: none"> <li>Same data as Models 8 and 9</li> </ul>	<ul style="list-style-type: none"> <li>ROI on technology innovation in one or more projects</li> </ul>

The materials in Appendix C, “Analyzing Models and Trends,” provides example planning and trending charts showing how the ISD Measurement Team uses software measures from projects.

### 3.3.2. Analyzing Process Compliance <sup>2</sup>

All projects that are required to adhere to the ISD process baseline will contribute information as to their adherence to compliance with the baseline. In Phase III, the information will be captured by way of process audits that occur on a periodic basis for each project.

The compliance information will be based on a scoring scheme for each of the functional processes in the scope of the baseline (e.g., planning, measurement, inspections, CM, QA, Requirements Management). The specific list of the process functional areas and the scoring criteria are defined in Appendix B.

---

<sup>2</sup> Note: this section describes a proposed approach for Phase III process compliance measurement and needs further development.

The measures computed will be in two forms:

1. Project-specific where the scoring will be used to record specific projects' compliance. This will be used as feedback to projects to help them make adjustments to their process use.
2. Project roll-up where the combined sampling will be used to characterize trends in the overall ISD process usage. This set of measures will be derived from data from all ISD projects.

The first analysis will be a representation of the process scoring sheet (Appendix B) and will be a rating for each of the process functional areas which were part of the project audits. It is anticipated that not all functional areas will be addressed in each audit (due to time) but over the life-cycle of the project it is anticipated that most process functional areas will be addressed. Each process function will then be rated on a percentage score of compliance derived directly from the scoring sheet. The second set of measures will be merely a roll-up of all audit results from all projects which participated in the audits. Several perspectives of this roll-up are generated, and this is dependant on the number and frequency of QLAs.

1. Trends in ISD process compliance: This is a quarterly summary of the overall scoring of process adherence for all projects audited during that time period. It is merely the effort-weighted average score for each of the process functions and the average score for the total number of processes which were examined.
2. Trends in specific process function adherence: The same type of trends can be generated for specific process functions (e.g., planning) as the scores are rolled up from all QLA information.

The procedure in Appendix D, "Analyzing Compliance Measures," describes the details of how the ISD Measurement Team uses this software process assessment information from projects.

The primary products from this work element are procedures and operational scenarios for data analysis.

The products generated in this work element will be updated, based on feedback from project users and the Deployment work element. Additional products may be generated, as necessary.

### **3.4 ISD Measurement Program Deployment Element**

The main functions of the Deployment work element are to infuse ISD-level measurement needs into project use and to support the projects in meeting their SPI-related measurement requirements. The key activities for deploying this program are seminars, project support meetings, periodic data-gathering interviews, and consulting for projects. Additionally, a set of support tools and materials describing the measurement program will be available.

The principal approaches of this work element include:

- Prepare measurement support materials and prepare SPI team.

- Deploy measurement plan to projects

Before initiating the direct deployment to projects within ISD, two steps will be taken by the SPI team. First, a set of support materials will be generated and packaged by SPI: this will include measurement examples, guidelines, suggested operational scenarios, and other aids that may be of use to projects adopting measurement activities. Additionally, a measurement deployment training seminar will be developed to be used to prepare any SPI team member who may be called on to participate in the deployment activities. This seminar will describe overall approach and support steps that any team member will take in working with projects. The instruments that will be used to support the deployment of the measurement program include management meetings, seminars, mentoring, and interactions during data-gathering interviews:

- The SPI team will brief ISD managers as to key elements of the measurement program and they in turn will reinforce the need for measurement to other project staff.
- Short seminars will be prepared and given to all project staff within the scope of the ISD process baseline. These seminars will describe the concepts, approach and required participation of the projects in carrying out successful measurement on their project.
- Mentoring will be provided by SPI team experts to any projects requesting support. The mentoring will provide guidance, suggestions and general information relating to the measurement program. The mentors will provide guidance in establishing and operating a measurement activity for a project, but they will not provide operational support.
- Interactions during data-gathering meetings will also help in the deployment of effective measurement. The person doing the data gathering will have access to examples, a list of other projects that have used or are using the measurement tools and general guidance that will also support the projects in adopting the measurement program.

The main products to be generated by this work element are:

- Support materials for software leads and other SPI Project teams (includes examples, guidance, and general tips on successful measurement)
- Briefing package for training all SPI mentors
- Briefing package for measurement seminars

The products generated in this work element will be updated, based on feedback from project users and the Deployment work element. Additional products may be generated, as necessary.

#### **4 ISD Measurement Program Interfaces and Organizational Structure**

The SPI Project will be responsible for operating the ISD Measurement Program. This will entail collecting, archiving, analyzing and reporting on ISD measures. All ISD measurement data will be archived electronically, whether through Excel spreadsheets, an Access database, other common electronic forms, or some combination of these elements.

The *measurement team* is responsible for:

1. Determining (in conjunction with Branch managers) which projects are in the scope of the ISD Measurement Policy.
2. Meeting with software manager at the beginning of a project to agree on definitions and collection approach, to collect initial data and to offer tools and additional services to the project.
3. Reviewing and quality assuring data provided by projects, and contacting projects to correct any errors or omissions.
4. Reviewing development completion data with software project manager to assure that correct information will be used to generate ISD software models.
5. Developing and maintaining the ISD measurement repository.
6. Populating the repository with data and reports collected.
7. Analyzing ISD measures to create planning models and to assess trends in productivity and quality.
8. Providing measurement-related process assets (e.g., data capture procedures, measurement guidelines and tools) to other SPI Project teams for use in their process definition and deployment efforts. These assets include a matrix of ISD project-level measurement practices mapped to NASA software engineering requirements.
9. Providing measurement support to individual projects as requested. This includes working with individual projects on how to collect and analyze project management metrics of interest as well as providing other reports to individual projects.

The *project development lead* is responsible for

1. Providing ISD measurement data according to an agreed upon schedule.
2. Act as the interface between the ISD and mission project management.

*Division and Branch managers* are responsible for

1. Establishing the ISD measurement policy and supporting the goals of this program.
2. Assuring that ISD software projects are meeting their responsibilities.
3. Reviewing the results of this program and recommending improvements.

*Other SPI Project teams* are responsible for

1. Incorporating ISD Measurement Team guidance into their process definition and deployment efforts.

## **5 Review, Reporting, and Metrics**

The status of the ISD Measurement Program effort will be reported at the SPI Project MMR.

Three categories of metrics will be developed and tracked to measure the impact that the ISD Measurement Program (MP) has on software development projects. These are:

1. Milestone Metrics – Quantitative type of metrics used for measuring the progress of the program itself. (e.g., adherence to schedule).
2. Infusion Metrics – Quantitative type of metrics used to track numbers of projects using ISD MP-generated products (e.g., number of projects reporting ISD measures, number of projects using ISD planning models).
3. Return on Investment (ROI) Metrics – Either qualitative or quantitative types of metrics used to measure the effectiveness of the ISD MP (e.g., management performance trends over time).

Metrics will be reported at reviews when they are collected and/or analyzed.

## **6 Project Schedule**

TBS

## **7 Budget**

TBS

## **8 Use of External Organizations**

The GSFC ISD MP will seek technical support from external organizations that have histories of expertise in measurement and analysis. To date, two such organizations have been engaged: Computer Sciences Corporation (CSC) and Fraunhofer Center-Maryland (FC-MD). Contracts with other organizations may be established if felt to be beneficial.

## **9 Risk Management**

A list of potential barriers and risks for the SPI ISD Measurement Program have been identified

**Table 2. Potential Barriers/Risks and Solution for Mitigation**

<b>Barrier/Risk</b>	<b>Risk Mitigation</b>
Gaining confidence and buy-in from ISD managers and projects	Update and evolve the plan based on bi-annual feedback and lessons-learned reports.
Managing expectations as to time required to realize benefits	Conduct periodic briefings on program results and data needs, showing how measurement data is being used.
Over commitment by SPI team	Periodically determine the need to rescope the plan and/or modify resources based on status reports on the SPI Project elements, including measurement team progress and deliverables.
Sustaining sufficient resources for measurement program's data capture and analysis activities	Plan for first results (e.g., in models, compliance assessments) within six months to demonstrate value-added to ISD.

## **APPENDIX A: Gathering Organizational Measures**

### **Introduction**

This appendix describes the minimally expected interactions between software projects and the ISD measurement team. These interactions include meeting with software projects to provide information on available measurement resources and to agree on any further support the ISD measurement team will provide. In addition to these scenarios, this appendix provides definitions of terms related to data collection.

### **Project Startup Scenario**

In this scenario a project is typically an ISD software development team, but the ISD measurement team is also prepared to support measurement programs being run at the branch level. The SPI project will provide a measurement expert to support either of these scenarios. In either case, the measurement team will have a set of two brief initial meetings with the project to define an appropriate, useful approach to measurement and to agree on mechanisms for collecting ISD-level data.

The initial meeting includes:

1. A discussion of project goals and objectives, and where measurement fits into attaining them.
2. Agreement on how data are to be collected, and how the standard ISD spreadsheet is tailored (if necessary) for the project. A typical tailoring of the spreadsheet would be to plan data collection at a single design review for small projects that aren't conducting separate PDRs and CDRs.
3. A brief review of the measurement requirements stated in NPR 7150.2. This discussion will emphasize the choices of measures that meet the requirements, and how a project can analyze these data to produce useful information to the PDL.

The follow-up meeting includes:

1. A presentation by the measurement expert of recommendations on a measurement approach to meet the software project's goals and objectives.
2. The software project and measurement expert agree on any additional services from the ISD measurement team, if any.

Table 1 below shows the project characteristics information; these data are provided at this point, with the exception of size data, which is provided at the end of the project. See Table 3 for definitions of these data items.



<b>Project Name</b>	
<b>Contact Name</b>	
<b>Contact e-mail</b>	
<b>Software Type</b>	

CSCI Name	Class	CSCI Type	Language(s)	COTS Product(s)	HW	OS	Size	Units

**Table 1. Project Characteristics Data**

In addition to collecting project characteristics at the start of a project, the beginning of a project is also a milestone. Thus we also collect an initial set of milestone data as described in Table 2 below. NOTE: The start of a software project is defined as the point at which one starts recording staff time for a product development team. Estimates of cost and schedule should be consistent with this definition.

### **Milestone Data Collection**

As the name implies, milestone data are collected at a few key points in a software project; the start of a project, design reviews, the start and end of testing, and the delivery to a customer for operations and maintenance. Table 2 below shows the data collected at each milestone; one column is filled out at each milestone. The gray fields represent actual data; the white fields represent estimates. Definitions for data items are presented in Table 4.

In addition to these data, the ISD team will capture the basis of estimate for the data, and narrative information describing any changes in estimates since the last time data were provided.

In any of these data fields, a value and a note identifier can be entered. Identifier text can be entered to provide additional commentary, such as rational for values, or describe project events driving change.

Project Name	
Current Date	

<b>Event:</b>	<b>Start</b>	<b>SRR</b>	<b>PDR</b>	<b>CDR</b>	<b>Start Test</b>	<b>End Test</b>	<b>Maint.</b>
<b>Basis of Estimate Provided? (yes/no)</b>							

#### Estimated and Actual Milestone Dates

System Requirements Review							
Preliminary Design Review							
Critical Design Review							
Start of System Testing							
Acceptance Test End							
Turnover to Maintenance							

#### Progress points (from point counting)

Actual at milestone							
Estimated at completion							

#### Effort (expressed as FTEs)

Actual at milestone							
Estimated at completion							



#### Requirements Data

Number of requirements							
Number of TBDs							
Cumulative changed requirements							

#### Cumulative Defects

critical defects found							
moderate defects found							
minor defects found							

#### Color Coding Key

Gray fill = actual values -   
 White fill = estimates - 

**Table 2. Milestone Data**

The data collected are expected to be values effective at the milestone dates, however the data are provided on a schedule agreed to at the initial meeting between the measurement team and the software team.

## **Project Closeout**

The end of a development project is defined as the turnover of the system to a customer for maintenance and operations. At project closeout the data capture involves filling out the last column of the milestone data and the size information needed to complete the project characteristics. The size information includes unit of measure. It is up to the ISD measurement team to convert size to a common unit of measure that allows cross-project collaboration.

At closeout, the measurement team will perform an “exit interview” with the PDL to capture key project events. This interview is designed to help the team correctly interpret the measurement data that the project has provided.

## **Data Definitions**

The following tables provide definitions for the data items discussed in this Appendix. Table 3 contains definitions for the Project Characteristic data listed in Table 1. Table 4 provides definitions for the Milestone Data of Table 2.

<b>Project Characteristic</b>	<b>Definition</b>	<b>Notes</b>
Project Name	e.g., ST-5 Flight Software, EOS Telemetry & Command System	
Contact Name	Name of the person within the software project with whom the Measurement Team interacts.	This is usually the PDL or his/her designee.
Software Type	The high-level domain of the software being developed or maintained.	ISD domains are: flight, ground, science analysis, infrastructure, and research.
CSCI Name	The short name or acronym for each computer software configuration item (CSCI) to be delivered by this software project.	For ISD, a CSCI is defined as a subsystem and platform combination.
Class	The software classification (A through H, per NPR 7150.2) of the CSCI.	
CSCI Type	The type of software being developed or maintained.	Examples of CSCI Types include: (For FSW) C&DH, ACS, and SI; (for Ground SW) T&C, P&S, FD, SD Lv1, SD Analysis, and Engineering Trending.
Language	The programming language(s) in which the CSCI is written.	
OTS Products	A list of off-the-shelf products incorporated into or delivered with the CSCI.	Includes Commercial off-the shelf (COTS), Government off-the shelf (GOTS) and Modified off-the shelf (MOTS) products.
Hardware (HW)	The processor for which this CSCI is targeted.	E.g., Power PC (RAD750)
Operating System (OS)	The operating system used by this CSCI.	E.g., RTEMS, VxWorks

Project Characteristic	Definition	Notes
Software Size	The size of a delivered CSCI, expressed in units such as source lines of code, number of modules, etc.	The sum of all CSCI sizes should equal the total size of the software system.
Units	The units in which software size is expressed.	

**Table 3: Definitions of Project Characteristic Data**

Measure	Definition	Notes
Effort	Staff-years expended on the software project. Includes software systems engineering, development, integration, testing, management, and other support.	<ol style="list-style-type: none"> <li>1. Management effort should be recorded up to and including the levels of PDL and Line Manager.</li> <li>2. "Other support" includes training, documentation, and user support.)</li> <li>3. Staff-years should be expressed as Full-time Equivalents (FTEs) that account for vacations, holidays, etc.</li> <li>4. Effort can be actuals or estimates</li> </ol>
Progress points	Points earned for execution of project activities or completion of deliverables.	Progress points are estimated and recorded by the PDL using a Point Counting Spreadsheet or other tool.
Defects	The number of discrepancies (i.e., errors) found in software products that are under configuration control. Defects are collected in three severity levels: critical, moderate, and minor.	<ol style="list-style-type: none"> <li>1. A "critical" defect prevents test team progress or operational use of a system/subsystem, build, or release. No work-around is possible or practical.</li> <li>2. A "moderate" defect prohibits successful completion of one or more test variations. It is serious but does not prevent using or testing a required capability.</li> <li>3. A "minor" defect does not prohibit successful completion of a test. This category involves minor deviations from task or project standards.</li> </ol>
Milestone dates	Date for each of the major milestones identified by the project.	<ol style="list-style-type: none"> <li>1. The nominal set of Milestone Dates is Project start, Software Requirements Review (SRR), PDR, CDR, start of system testing (e.g., STRR), completion of acceptance testing (e.g., Acceptance Test Results Review), and turnover to maintenance.</li> <li>2. For small projects (e.g., &lt;5 FTEs) the minimal set of Milestone Dates is Project start, PDR (if held, otherwise SRR), CDR, and completion of acceptance testing.</li> <li>3. Large projects may report additional milestones, e.g., for Build CDRs and releases.</li> <li>4. Project start is defined as the date on which effort hours begin to be expended on the project by any member of the PDT, including the PDL.</li> </ol>
Number of requirements	The total number of detailed (i.e., testable) requirements listed in a software project's Requirements Traceability Matrix.	
Number of requirements TBDs	The total number of detailed requirements in a software project's Requirements Traceability Matrix that are listed as "To be determined (TBD)" or "To be supplied (TBS)" or are otherwise incomplete.	

Measure	Definition	Notes
Cumulative changed requirements	The sum of the number of additions, changes, and modifications made to detailed requirements.	

**Table 4: Definitions of Milestone Data**

---

## **APPENDIX B: Gathering Compliance Measures <sup>3</sup>**

TBS

This appendix is expected to contain two items (similar to examples Frank McGarry provided in his March 14<sup>th</sup> e-mail):

QLA procedure

QLA scoring spreadsheet

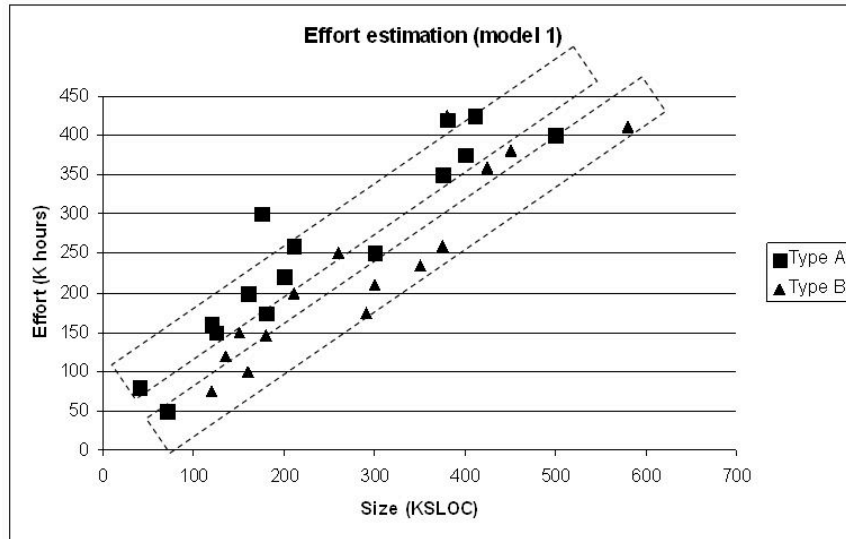
---

<sup>3</sup> *Note: this section needs to be provided in concert with the GSFC SQA organization*

## APPENDIX C: Analyzing Models and Trends

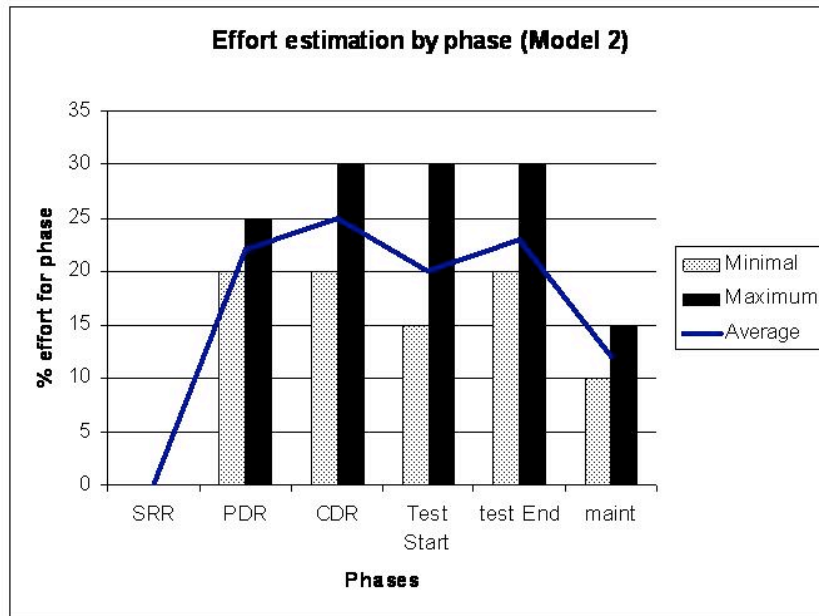
In the following sections, sample plots are presented of the models listed in Section 3.3.1. These are annotated with an explanation of how it can be used. These are only preliminary graphs. The real power of measurement is the ability to combine some of these graphs in new ways to learn more about a project. Note that all data in these graphs are fictitious.

### 1. Total effort estimate



From data collected at the end of a project, the total size and total effort for each project can be displayed. Type A projects (■) and Type B projects (▲) are displayed separately indicating a different relationship between the sizes of these projects and effort required to build them. This can be used as a preliminary check on early management estimates for a new project. Does it fall within the guidelines of previous projects?

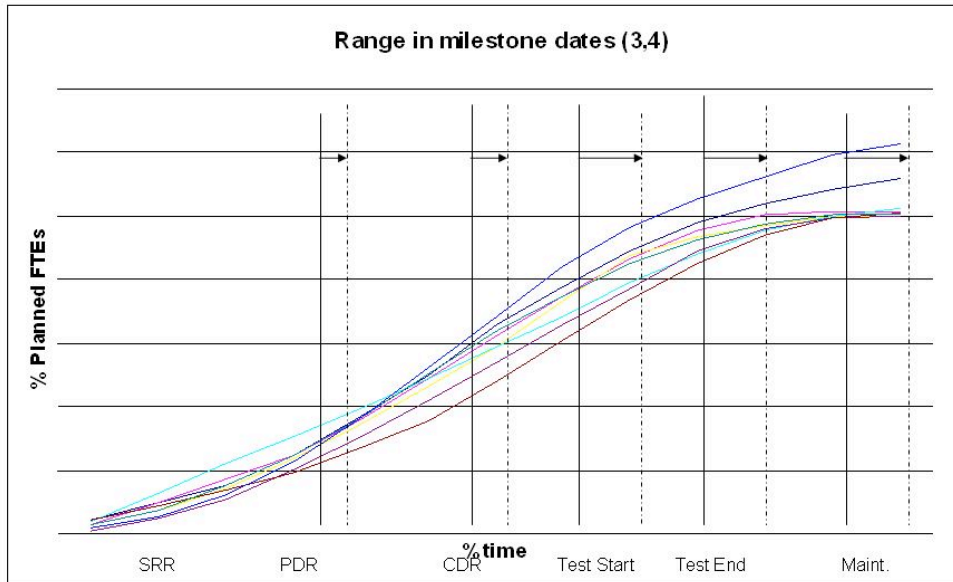
## 2. Effort estimate by phase



From historical data, the maximum and minimum effort for each phase can be plotted as well as the average percent of effort for each phase. This is a further check on effort breakdown for a new project. If new projects do not fall within these bounds that may either indicate an error in planning or that a special circumstance is present for this project. This chart provides a warning signal for a project manager in planning for a new project.



### 3. Total schedule estimates

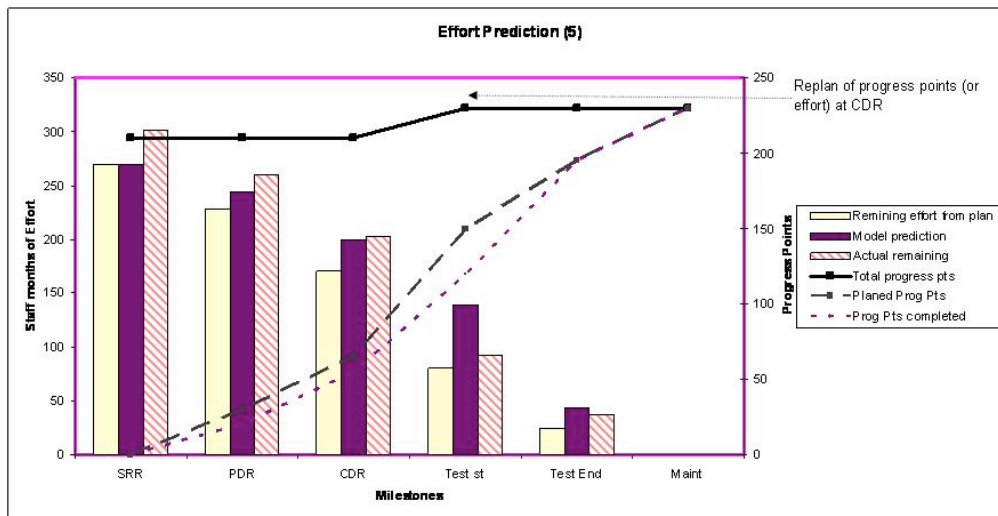


Plotting percent of effort completed at each milestone date, a set of characteristic curves can be plotted. By plotting both the scheduled milestone review date and the average slippage of these milestone dates, baseline profiles can be computed to determine average effort for each phase.

### 4. Schedule estimates by phase

Using the same plot as the previous graph, slippage in milestone dates can be computed. This allows baselines to be set for scheduling reviews on future projects and for determining if the proposed schedules on future projects are reasonable.

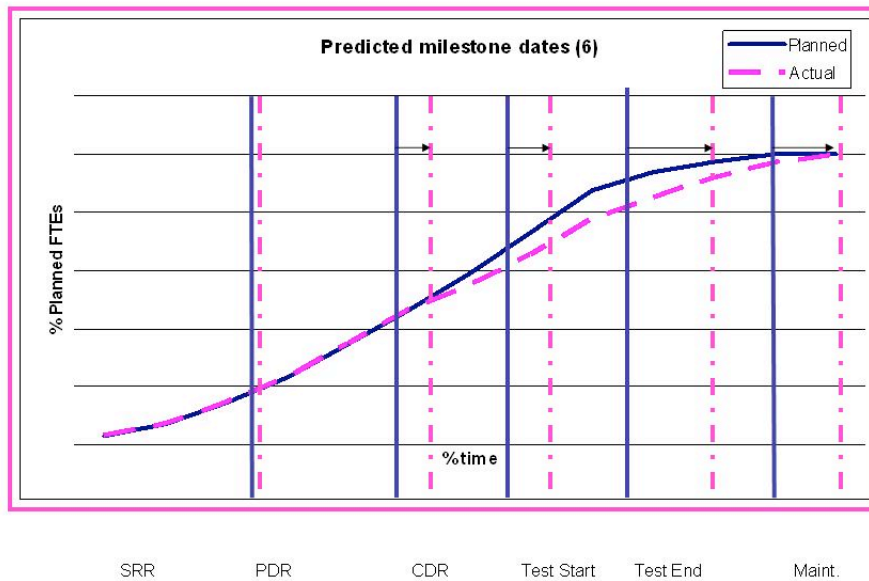
## 5. Effort prediction



In this graph, the left bar is the planned effort remaining in a project based upon the original project plan. The line at the top is the total number of progress points for the project and the dashed line represents the progress points scheduled to be completed by each milestone.

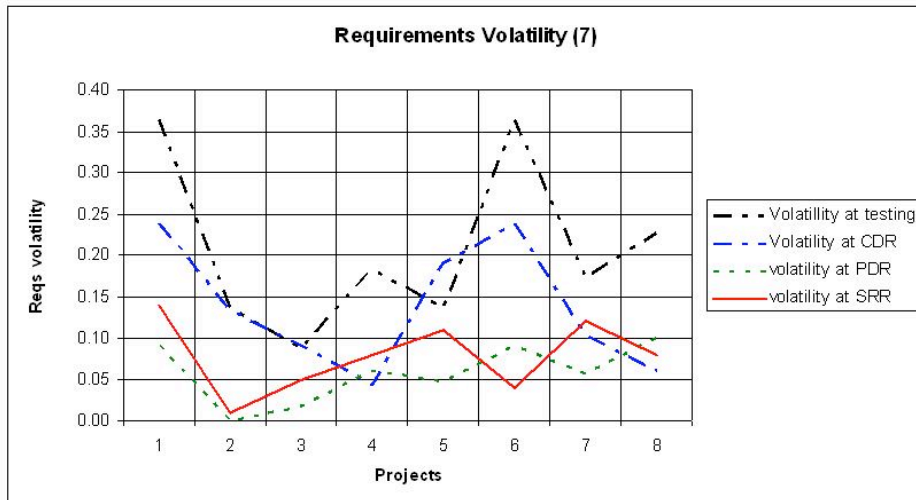
The dotted line represents the progress points actually completed at each milestone. In this example, fewer progress points were completed by PDR, CDR, and Test start, indicating that the project is behind schedule and will take increased effort to finish (center bar). The rightmost bar represents the actual effort to finish the project (determined at the end of the project) indicating that the prediction was indeed in the right direction (i.e., indicating a cost overrun early in the project). As experience with progress points increases, the accuracy of this prediction will also increase.

## 6. Schedule tracking and prediction



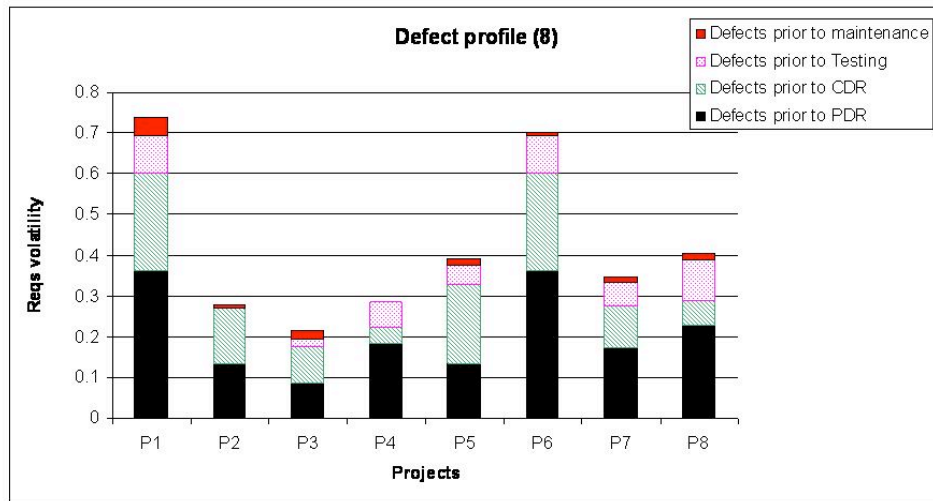
By keeping track of planned milestone dates (solid line) and actual percent of effort completed (perhaps adjusted by the effort prediction model above), the delay in milestone dates can be estimated. Given the data from model 2, we would know how much effort is typically required to do the remaining phases of the project at any given milestone and can be used to adjust the schedule for a new project.

## 7. Requirements volatility



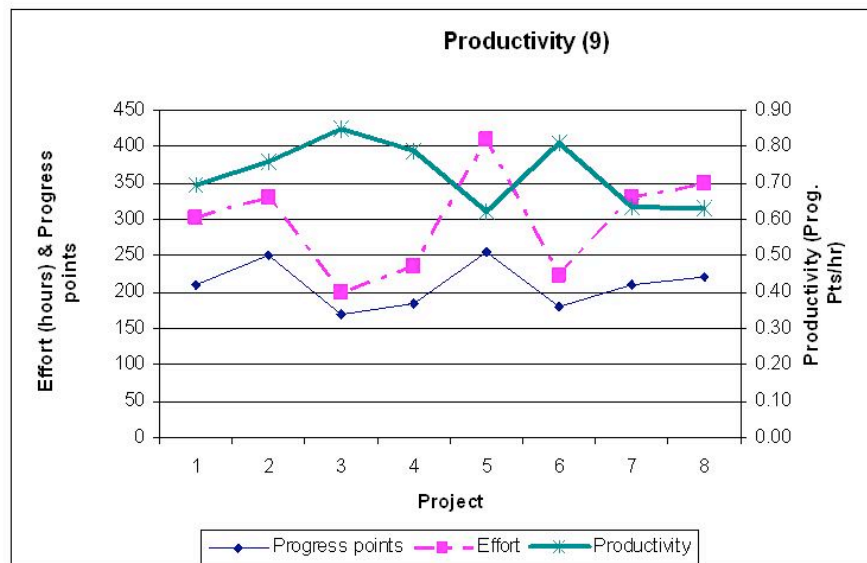
In this graph, the requirements volatility is plotted by phases across eight different projects. Comparing this with defects found, productivity, etc., gives a measure of measuring the impact of requirements volatility on quality and cost.

## 8. Defect profile



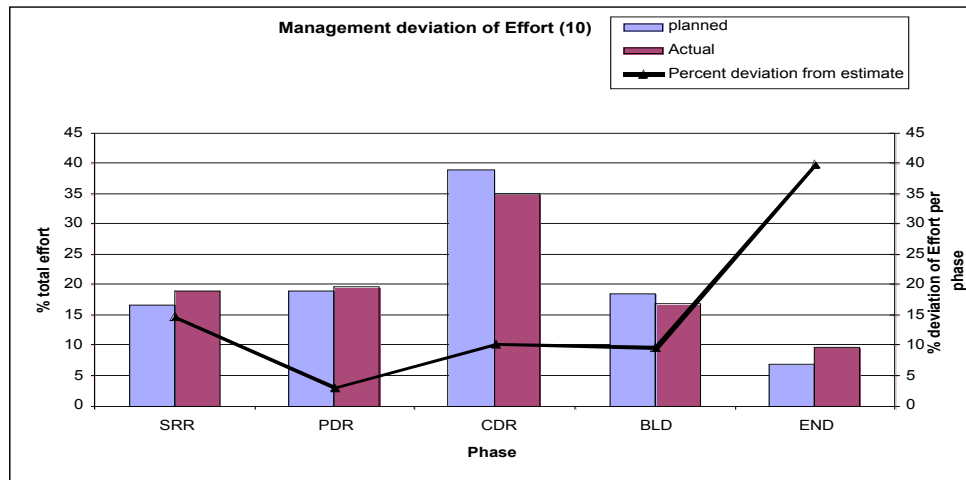
This graph displays the number of defects found in a project at each milestone across eight different projects. The value of these data, along with requirements volatility, is to compare these graphs with measures of productivity and quality.

## 9. Productivity

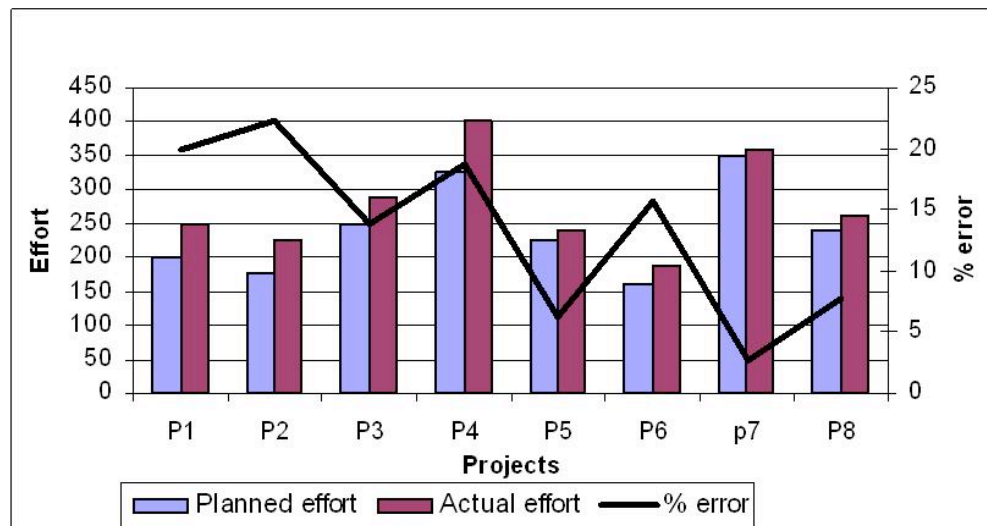


In this plot, productivity (as measured in progress points per staff hour of effort) is plotted with effort and progress points completed across eight projects. As with the last few charts, the value of this is in comparing productivity against defect and other reliability measures.

## 10. Management performance trend

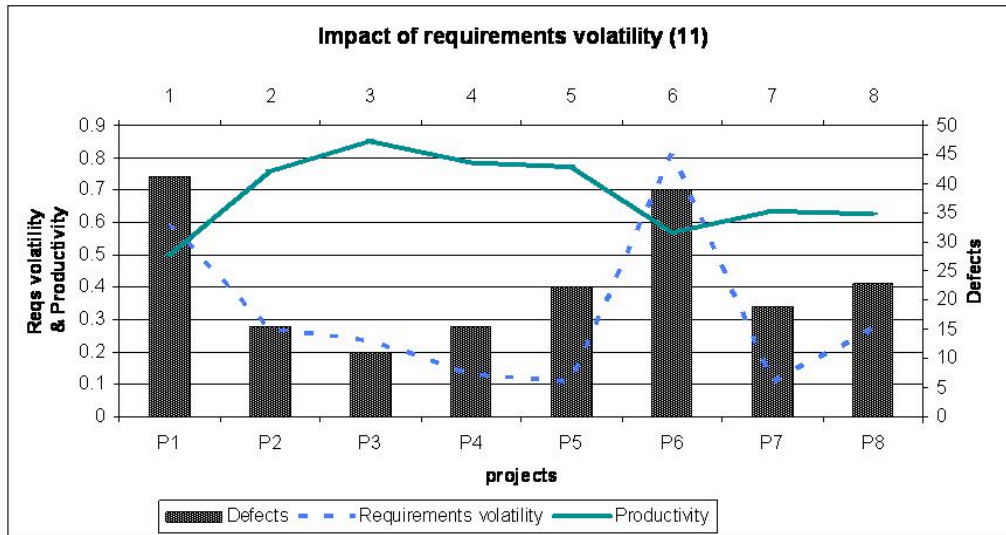


In this plot, the planned versus actual percent of effort for each phase of development is plotted along with the average percent deviation in each area. Much like plot 8 (defect profile), if total planned and actual effort is plotted as well as percent deviation, then management performance over time can be computed, as given below.



This graph demonstrates that over eight projects, estimating ability is increasing with a general trend downward in percent error in effort estimation.

## 11. Impact of requirements volatility



This plot shows the relationship between defects (bars), requirements volatility (dashed line) and productivity (solid line). It is expected that the data will show that requirements volatility is strongly correlated with increased level of defects and inversely proportional to productivity.

## 12. CMMI and technology charts

These will all look like graph 8 – a series of bars, where each bar represents a different characteristic being measured. These graphs need further development of the actual CMMI and technology factors that are to be measured.

## **APPENDIX D: Analyzing Compliance Measures <sup>4</sup>**

TBS

---

<sup>4</sup> *Note: this section needs to be provided in concert with the GSFC SQA organization*



## **APPENDIX E: Acronyms and Abbreviations**

ACS	Attitude Control System
C&DH	Command and Data Handling
CDR	Critical Design Review
CMMI (SM)	Capability Maturity Model Integrated
CSC	Computer Sciences Corporation
CSCI	Computer Software Configuration Item
FC-MD	Fraunhofer Center – Maryland
FDS	Flight Dynamics System
FSB	Flight Software Branch
GSFC	Goddard Space Flight Center
ISD	Information Systems Division
JPL	Jet Propulsion Laboratory
Lv1	Level 1
MP	Measurement Plan
MT	Measurement Team
NASA	National Aeronautics and Space Administration
NPR	NASA Procedural Requirements
P&S	Planning and Scheduling
PDL	Product Development Lead
PDR	Preliminary Design Review
QLA	Quick Look Analysis
SEL	Software Engineering Laboratory
SD	Science Data
SI	Science Instrument
SPI	Software Process Improvement
SQI	Software Quality Initiative
T&C	Telemetry and Control